```
%ESTIMATION CODE - Importance Sampling  - Weitzman, UrsuSeilerHonka 2022
clc; clear all;

tic
%options for estimation
options = optimset('Display',
'final','DiffMinChange',0.05,'FinDiffType','central','FunValCheck','on','HessUpdate
', 'bfgs', 'MaxFunEvals',6000000,'MaxIter',6000000,'TolX',10^-6,'TolFun',10^-6);


%--------------------------------------------------------
% Setting parameters (obtained from file name)
%--------------------------------------------------------

filename=mfilename;

%number of epsilon+eta draws
D=str2num(filename(15:17));

% Num of proposal simulations
M = str2num(filename(19:21));

%seed
seed=str2num(filename(23:end));

%--------------------------------------------------------
% Simulation
%--------------------------------------------------------

%simulation inputs
N_cons=1000;%num of consumers
N_prod=5;%num of products
param=[1 0.7 0.5 0.3 -3];%true parameter vector [4 brandFE, search cost constant
(exp)]
param_std = 0.1.*ones(size(param));

%simulate data with random coefficients
simWeitz(N_cons, N_prod, param, param_std, seed);
%load simulated data
data=load(sprintf('genWeitzDataS%d.mat',seed));data=cell2mat(struct2cell(data));

%--------------------------------------------------------
% Estimation
%--------------------------------------------------------

% Proposal density parameters
param_mean_prop = param;
param_std_prop = 0.1.*ones(size(param));
param_prop = [param_mean_prop param_std_prop];
```

```
% Initial guess
param_mean0 = zeros(size(param));
param_std0 = ones(size(param));
param0 = [param_mean0 param_std0];

%Maximization
f = @(x)liklWeitz_imp_1(x, param_prop, data, D, M, seed);
%problem has no linear constraints, so set those arguments to []
A = [];
b = [];
Aeq = [];
beq = [];
%upper and lower bounds on the parameters
lb = [-4 -4 -4 -4 -4 0 0 0 0 0];
ub = [4 4 4 4 4 2 2 2 2 2];

est = imp_sampling(param0, param_prop, data, D, M, ub, lb, seed, options);

%save results
csvwrite(sprintf('rezSimWeitz_imp_D%dM%dS%d.csv',D,M,seed),est);
toc
```